(19) FEDERAL REPUBLIC OF GERMANY

GERMAN PATENT OFFICE

(12) **Unexamined Patent Application**

(10) **DE 196 02 804 A1**

(51) **Int. Cl. 5:**
G 11 B 23/28
G 06 F 12/14

(21) File Number: 196 02 804.3
(22) Date of Filing: January 26, 1996
(43) Date of Publication: July 31, 1997

(71) Applicant:
Harras, Roland, 82031 Grünwald, DE

(72) Inventor:
Inventor will be named later

(56) References
DE 1 95 10 436 A1

Request for examination is filed in accordance with § 44 of the Patent Law.

(54) Method for preventing the copying of software (software copying protection)

(57) Software (multimedia, video films, computer data or programs) is relatively easy to copy during use or prior to it. This option is used by organizations as much as by individuals for producing illegal copies and/or passing on (multiple copies of) software. With the help of the invention, undesirable copying is prevented and/or pirated copies become unusable.

The method according to the invention is used to first write traceable small data units on a data medium (supplies the software) in a certain region and then consciously damage or modify precisely this region. The resulting effects on the data units are analyzed and documented together with the software that is to be protected. During subsequent applications of the software, the corresponding data medium can be re-analyzed and the results can be compared to the documented results to verify the authenticity of the data medium. An exact reconstruction of the damage/modification of the data medium, which would produce exactly the same errors, is impossible.

Description

Particularly in the present age, pirated copies of various software programs represent a great problem for manufactures and for associated sales organizations. This is particularly evident when it comes to video, audio, computer programs, data and multimedia. In light of the development of re-writeable CDs with very high storage capacities, MODs, MDs etc., the problem, which has so far been observed most frequently with diskettes and video tapes, will probably increase and expand even further in the future. It is difficult to estimate the damage.

Existing efforts, such as the request for a serial number in the case of computer programs or the modification of an installation disk during installation or the installation of signals on video tapes to make the copy unusable, have hardly improved the situation. Professional organizations, and often even individuals, were still able to produce operational copies.

Thus the invention provides a solution with the following method, which for ease of understanding is illustrated based on computer software.

1. Extremely small files are written (in part) on a data medium. The data, configuration and descriptions of these files are always the same and traceable. If, for example, a computer program is supposed to be protected, this data medium could also be one of the (plurality of) installation disks. In the event of smaller programs (or when the data medium is a CD or a medium with similarly large capacity), the files could also be written on about 1/3 of the data medium and then the program itself could be provided on the remaining 2/3. However, in this case it should be ensured that the actual program is not affected by the subsequent damaging of the data medium.
2. Then, the data medium is intentionally damaged/modified in the region on which the above files have been written. This could occur with laser, a hot, pointed object, paint, or simply by a scratch produced with a scalpel. This damage and above all the shape, size and locality thereof should be defined randomly. The modification should ideally be carried out manually, or by means of a machine with a random number generator. It is also possible to damage the medium in multiple locations. The only important aspect is that the damage occurs exclusively in the region of the data medium to which the corresponding test files were written in accordance with number 1. If programs and/or other data are also to be accommodated on the data medium, these are not to be damaged.
3. Thereafter, a scan program attempts to read the files/data written according to 1. In the damaged locations, this will create problems, or the data will be reproduced in a modified fashion/incorrectly. This typically produces a sort of read error, which is known to occur on disks. With "Windows", generally the operating system would even abort the scan program by indicating a system error such as "Cannot read from the data medium", however the scan program suppresses this effect.

Thus, the effects outlined for the modifications carried out according to 2, are being recorded and stored. This step will function like a fingerprint of the data medium, and as such will differ from one data medium to another. Furthermore, this identification cannot be copied since during an attempt to do so the copy process would be aborted. Even with special copy programs, a duplicate of the data medium could only be produced insofar as sort of a "blank space" would exist on the copy in all the damaged locations. This, however, would not produce the above-mentioned read error during future scans.

4. The results of the analysis gathered and stored/documented according to 3 are now "hidden" in the program (usually in the main program, not in the resource) and/or in the software that is to be protected and that is supplied with the data medium. The results could also be accommodated in a non-manipulated region of the data medium, preferably in encrypted form. Maximum security is achieved when the program is not compiled until after these results have been inserted. In any case, in this manner this software alone is connected to this data medium alone. They are quasi "bonded" forever.
5. The software to be protected has a test/scan program, which is activated every time the application/software is started or during every use of the software. This test program asks the user, for example, every 14 days to insert the modified data medium in the drive used during installation. It will then test the data medium as described above according to 3. This way, referenced information is obtained, which can be compared to the data documented in accordance with 4.
6. If the data, outside of certain tolerances, should not coincide, it is certain that this is not the original data medium. In this case, the test program could then erase the respective software or simply refuse further access to this software.

The essential aspect is that the damage/modification of the data medium carried out according to 2 cannot be exactly reproduced (or at least is extremely unlikely). Also the chance in general of an exact copy of the data medium is quite low. After all, reading of the destroyed regions of the data medium is not possible (or is garbled). This produces a read error. Most operating systems consider this to be a system error, which cannot be reproduced with software or a copy device. It is very likely instead that the file structure on the new (copied) data medium will be different than on the original. However, in any case, the new data medium can only produce error messages such as "File cannot be opened" or "File not found" or "Data error". The test/scan program (see numbers 3 and 5), however, inquires precisely about the error messages occurring after physical damage.

Thus, it is apparent that only the user who is in possession of the original installation data medium will be able to permanently use the associated software. Unlike the variation described below, however, the owner will be able to install the software on several computers in his area or house. Sometimes this is desirable and manufacturers allow this practice. It is also possible to integrate a sort of "buffer" in the test program, so that the original installation data medium does not absolutely have to be inserted following the first request. It is conceivable to give the user a "second chance" (or, of course, a third one).

It is also conceivable to delete or modify a certain file (or a plurality of files, or also the entire content) during the installation of the software, this modification prompting the same installation program during the subsequent attempt to install the software on a (probably different) computer not to carry out any further installations. The inability to copy the installation data medium or one of the installation data media therefore would exclude the common practice of copying the installation data medium prior to the installation and then passing it on.

A test/scan program, which was written for Windows 3.x and serves for the protection of computer software, is available from the inventor. This would allow the production, documentation and subsequent testing of these special data media (with the exception of intentional damage).

In the near future, surely the writeable compact disk (successor of the existing CD, or CD-ROM) will gain importance. It will replace, above all, the video tape (VHS, Video 2000, Beta, V8) and also dominate the remaining multimedia market. Particularly in the video film sector, this will further worsen the existing tremendous problem of pirated copies because copying will no longer be associated with a loss of quality.

In this case, for example, a test program that has been included on the CD could test the authenticity of the CD – which has been processed using the above method - prior to the start of the video film. Only if the test is successful will the video film be released for viewing. Due to the many different formats, perhaps also the 'playback program' could be delivered together with the video film. This program could then include the above test program.

Claims

1. A method for preventing the copying of software (software copy protection) that has been applied to a data medium and/or is delivered with/through a data medium or is connected to such a medium or secured by it, **characterized in that** one or more intentionally produced physically damaged locations or modifications of the data medium are analyzed precisely in terms of their effects on data or files previously written/stored on the data medium and that the results are documented/recorded/stored such that, during use of the software to be protected and/or the installation thereof, a or the software and/or program or an electronic device once again analyses this data medium and compares the results to the documented results of the previous analysis in order to test the authenticity of the data medium and then responds accordingly.

2. A method for preventing the copying of software (software copy protection) that has been applied to a data medium and/or is delivered with/through a data medium or is connected to such a medium or secured by it, characterized in that one or more modifications/damaged locations of the data medium, which cannot be reproduced or are very unlikely to be reproduced and/or cannot be copied or are very unlikely to be copied, are created and documented such that they can be verified repeatedly through (computer) technology.

3. The method according to claims 1 and 2, characterized in that the data medium can be a disk or a compact disk in various sizes and formats (CD, SD, MOD, MD and any types developed in the future) and can be writeable and/or erasable one or multiple times. This also includes data media for video, audio, multimedia as well as variations thereof, including streamers, data tapes, cassettes and also hard drives in their respectively diverse variations.

4. The method according to claims 1 and 2, characterized in that the modifications and/or damaged locations are produced by one or more scratches, cuts, deformations, heat supply and/or thermal effects, laser applications, paints, applications of other substances, chemical changes of the surfaces, magnetic changes, changes to the magnetic or other properties, and/or changes of the optical properties, or by the addition or removal of materials.

5. The method according to claims 1, 2 and 4, characterized in that the modifications and/or damaged locations are lasting/permanent and constant to a certain extent – also over extended periods.

6. The method according to claims 1 and 2, characterized in that first data is written on the data medium such that the subsequent modification/damage is easy to locate/detect and/or analyze. It is also possible to define only a certain portion/region of the data medium that is supposed to be modified/damaged. It is furthermore possible to store other, method-independent data on the affected data medium, however this data would be limited most of the time to a partial region of the data medium, which does not serve this method and to which therefore the above data is not written, and which is likewise not subsequently damaged or modified in accordance with the above claims.

7. The method according to claims 1, 2 and 6, characterized in that the data is included in (many) small or (few) large, simple or complex files. It is also possible to directly write on/emboss/imprint/etch/burn blocks, sectors, tracks or other defined areas/parts of the data medium.

8. The method according to claims 1 and 2, characterized in that the modifications and/or damaged locations of the data medium and the respective effects thereof are analyzed and captured in that the previously applied data (claims 6 and 7) are read/tested and corresponding deviations from the original target values (prior to the modification/damage) or problems occurring (precisely due to this analysis program) can be accurately recorded/documented.

9. The method according to claims 1, 2 and 8, characterized in that the documentation of the modifications and/or the results of the analysis following the damage to the data medium are accommodated, or hidden, in a or the program/software to be protected, in a file, encrypted on the or a different data medium, or other "location" that is difficult to manipulate.

10. The method according to claims 1 and 2, characterized in that the data medium can be repeatedly tested in that the data medium and/or the data stored thereon are analyzed and/or read/tested in the same manner as it was carried out – as explained in claim 8 – for documenting the effects of the intentional modification/damage to the previously applied data and that the present results of the analysis are compared to the above documentation (of the first results of the analysis).

11. The method according to claims 1, 2, 8, 9 and 10, characterized in that the documentation (of the results of the first analysis) according to claim 9 is accommodated in a or the program/sub-program of the respective software (to be protected), in conjunction with a test program for testing the data medium, such that this test program demands, at certain intervals and/or randomly and/or after a certain application time/duration and/or a certain number of uses or a combination thereof, the modified/damaged data medium to then test it for authenticity, as has been illustrated according to claim 10. The above test program can also be integrated in a unit/drive/electronic device or be carried out by an electronic circuit.

12. The method according to claims 1, 2, 8, 9 and 10, characterized in that the documentation according to claim 9 is accommodated in a or the program or partial program or sub-program (preferably the installation program) of the associated software (to be protected), in conjunction with a test program for testing the data medium (in accordance with claim 10), such that this test program during installation asks for the modified/damaged data medium to then test it for authenticity and creates an annotation/a modification/deletion on this data medium so that this test program in the future is informed about the completed installation. The above test program can also be integrated in a unit/drive/electronic device or be carried out by an electronic circuit. As a result, multiple installations of the same program can be prevented and/or controlled.

13. The method according to claims 1 and 2, characterized in that the analysis of the data medium described according to claim 10 is carried out by a test program, which is integrated in the software to be protected as a partial program or sub-program of the main program or is accessed by the software to be protected as a separate program, or is integrated in the drive used to play the data medium, an additional device, the playback device, computer or other electronic device or replaced by it.

14. The method according to claims 1 and 2, characterized in that in the event an impermissible deviation of the tested data medium from the original is detected according to claim 10, the application and/or the software to be protected is blocked, destroyed, frozen or aborted, or the software to be protected is only decrypted when the tested data medium coincides with the original since it only exists in encrypted form and is useless without decryption, or the software to be protected is only loaded, started or played back when the tested data medium coincides with the original.